

# **DATA DELIVERY WITHIN THE INFORMATION WAREHOUSE FRAMEWORK**

*March 19, 1992*

IBM Programming Systems Headquarters Marketing  
Version 1.0

IBM Corporation  
Programming Systems Headquarters  
Route 100, Box 100 OBI/4A-14  
Somers, NY 10589

## Preface

The purpose of this document is to clarify the role of the data delivery element of IBM's Information Warehouse<sup>1</sup> strategy and to position the components of the Data Delivery element. It is intended to provide a short description of the components and product positioning and is **NOT** intended to replace the more comprehensive documentation and publications available from IBM, Bachman Information Systems, Inc., and Information Builders, Inc. that describe the Information Warehouse framework and products within the framework in more detail. All of the information concerning the non-IBM products in this document was supplied by the individual product vendors.

For a more comprehensive overview of the Information Warehouse framework and its components, please refer to the publication:

*Information Warehouse: An Introduction (GC26-4876)*

It is anticipated that employees of IBM and the members of the International Alliance for the Information Warehouse framework will use this document to assist customers in understanding the options available to them for Data Delivery within the Information Warehouse framework, and is **NOT** intended as a product tutorial.

Questions regarding the Information Warehouse framework should be directed to the local IBM marketing organization, or the IBM support organization within your geography.

Suggestions for improving this document should be sent to:

IBM Corporation  
Programming Systems Headquarters  
Route 100, Box 100 OB1 4A-14  
Somers, NY 10589

VNET = IBMMAIL(USIB26GC)

---

<sup>1</sup> Trademark or Registered Trademark, International Business Machines Corporation

# Table of Contents

<b>Introduction</b> .....	<b>1</b>
<b>Information Warehouse Framework Elements</b> .....	<b>2</b>
Enterprise Data .....	2
Data Delivery .....	2
Applications and Decision Support Systems .....	2
<b>The Data Delivery Element</b> .....	<b>3</b>
Data Access .....	3
Accessing Relational Data .....	3
Accessing Non-Relational Data .....	3
Concurrent Access to "Mixed" Data Sources .....	4
Data Movement and Enhancement .....	4
Data Extract Utility (DXT) .....	5
Data Propagator (DProp) .....	5
<b>Positioning</b> .....	<b>6</b>
<b>Notes on the Diagram</b> .....	<b>7</b>
Access to Relational Data (Paths 1 & 2) .....	7
Access to Non-Relational Data (Path 3) .....	7
Access via EDA API/SQL (Path 4) .....	7
<b>Summary</b> .....	<b>9</b>

# Introduction

Most customers have to deal with the ever increasing complexity of managing the storage and retrieval of data for transaction and Decision Support Systems (DSS) within their organization, since data is stored in various database and file systems which often use different access methods. The introduction of relational database technology by IBM added another data management structure, and a new access language, the Structured Query Language (SQL). In the early days of relational technology, it was used primarily by the knowledge worker to gain access to data for decision support applications. With the increases in function and performance that IBM has delivered for relational database across the IBM SAA<sup>1</sup> platforms, customers began migrating an increasing number and variety of applications to this environment. However, for various reasons, much of the data in the enterprise has not been moved to relational technology. As a result, customers continue to face the challenge of data management across the diversity of systems, tools, and languages while attempting to meet the requirement of making data available to knowledge workers in a usable form.

On September 11, 1991, IBM announced the Information Warehouse framework to address the customer requirement for providing data to knowledge workers regardless of its location or the database management or file system utilized to store the data (refer to Programming Announcement 291-471 dated September 11, 1991).

The Information Warehouse framework is comprised of three elements:

- Enterprise Data
- Data Delivery
- Applications and Decision Support Systems

There are numerous considerations when providing access to Enterprise Data for applications and Decision Support Systems that are beyond the scope of this document. The classic "who, what, where, when, why and how" questions must be answered when addressing the tools, data delivery, and data placement issues with continual attention to security, recoverability, function and performance. The selection and use of specific Enterprise Data, Data Delivery, and Applications and Decision Support System products within the framework is influenced by the decisions a customer will make regarding these considerations, and is tempered by:

- The customer's strategy for decision support
- The usability of the data in its current location and form
- The impact of the data access on existing applications
- The function available for the access path chosen (read-only / update)

IBM offers Consulting and Services to assist customers in dealing with these important aspects of the implementation of the Information Warehouse framework.

# Information Warehouse Framework Elements

## Enterprise Data

The scope of the Enterprise Data element includes data from most sources:

- local and remote
- relational and non-relational
- on IBM systems and non-IBM systems

The data within the Information Warehouse framework may be the operational data that is used by other applications and transaction systems, or it may be data that has been summarized, abstracted, propagated, or enhanced specifically for use by the knowledge worker as part of the decision making process.

The "enhanced data" is created by copying, subsetting, and creating data derivations of the operational data within the Information Warehouse framework by using tools that are part of the Data Delivery element of the framework.

## Data Delivery

The objective of the Data Delivery element of the Information Warehouse framework is to provide access to all of the data in the enterprise. Since Data Delivery is the topic of this paper, a detailed discussion will follow the description of the Applications and Decision Support Systems element.

## Applications and Decision Support Systems

The third element of the Information Warehouse framework encompasses the Applications and Decision Support System (IDSS). Knowledge workers should not need to be concerned with the location or the technology used to store or retrieve the data they desire for their query, report writing, analysis, statistics, graphics, etc., but should be able to focus their attention on the problems they are trying to solve.

The Information Warehouse framework provides for a wide variety of Decision Support System applications and tools to access the Enterprise Data. These applications and tools can support the knowledge worker with decisions that may assist their organization in attaining a competitive advantage in their industry. Tools that use supported levels of the Structured Query Language (SQL) to access data, such as IBM's Query Management Facility<sup>1</sup> (QMF<sup>1</sup>), Application System (AS), and Data Interpretation System<sup>1</sup> (DIS) should be able to access the data within the Information Warehouse framework, both operational and enhanced data, as defined later in this document.

## The Data Delivery Element

The Data Delivery element of the Information Warehouse framework is designed to include the components needed to provide access to all of the data in the enterprise, and for the movement of data into relational databases.

### Data Access

Access is provided to original data, created and used by transaction systems, that is stored in relational and non-relational structures, in addition to enhanced, or informational data, used by the knowledge worker. Access to data within the framework is provided by a common language, the Structured Query Language (SQL), through a single programming interface, the SAA SQL database interface<sup>2</sup>. The interfaces used today to access the various data are as follows:

#### *Accessing Relational Data*

Access to both local and distributed relational data is via the SAA SQL database interface. Access to distributed relational data is based on the Distributed Relational Database Architecture<sup>1</sup> (DRDA<sup>1</sup>), IBM's open architecture for interoperability among relational database management systems in the multiple-vendor customer enterprise. DRDA is a robust architecture that provides full read/write access to distributed relational data with data integrity, security, recoverability, reliability, availability, and performance. DRDA was announced in June 1990, and is further described in IBM's September 11, 1991 announcement (Refer to programming announcement 291-471 for details).

DRDA has been implemented across the SAA platforms by DB2<sup>1</sup>, SQL/DS<sup>1</sup>, and OS/400<sup>1</sup> relational database managers, and by Distributed Database Connection Services/2<sup>1</sup> (DDCS/2) for OS/2<sup>1</sup>.

DRDA is a published architecture, and many leading software vendors have already announced their intent to support this architecture.

#### *Accessing Non-Relational Data*

Access to non-relational data is also provided via the SAA SQL database interface, in conjunction with Enterprise Data Access/SQL<sup>3</sup> (EDA/SQL<sup>3</sup>) and the EDA/Dynamic Extender<sup>3</sup> products from Information Builders, Inc. (IBI), an International Alliance member for the Information Warehouse framework. (Refer to programming announcement 291-498 dated September 11, 1991).

The EDA/Dynamic Extender for DB2 and SQL/DS provide applications written to the DB2 or SQL/DS programming interface read-only, dynamic syntax SQL access to over 50 heterogeneous data sources across 35 operating environments via an ANSI 89 Level 1 SQL subset.

---

<sup>2</sup> Sometimes called the SAA SQL CPI

<sup>3</sup> Trademark, Information Builders, Inc.

The Enterprise Data Access/SQL family of products also provides a callable SQL interface (EDA API/SQL) which may be used in two ways:

- A consistent read-only subset of the ANSI 89 Level 1 SQL may be used to access any EDA/SQL-supported data sources. The EDA API/SQL statements are translated into the necessary data manager calls by EDA/SQL, and the translated statement is sent to the supported data manager for processing.
- For relational data, any SQL statement may be "passed through" EDA/SQL directly to the supported database manager.

### ***Concurrent Access to "Mixed" Data Sources***

When necessary, data from different data stores that reside on the same subsystem may be accessed and joined using a read-only subset of ANSI 89 Level 1 SQL via the SAA SQL database interface in conjunction with the EDA/Dynamic Extender for DB2 and SQL/DS, or via the EDA API/SQL.

### **Data Movement and Enhancement**

Enhanced data is data that is obtained from the original data store that may have been summarized, aggregated, abstracted, propagated, or enhanced specifically for use by the knowledge worker before being placed in an IBM SAA relational database. Customers are assisted with many of the database administration functions related to deciding what data is to be enhanced, and the activities associated with the creation of the relational tables, by tools from Bachman Information Systems, Inc., an International Alliance member for the Information Warehouse framework. The BACHIMAN/Analyst Capture<sup>4</sup> offerings serve to capture existing data definitions from programs and other data definitional sources such as: IMS/DB, COBOL and PL/I file definitions, and IDMS<sup>5</sup>. New relational database designs can be structured and databases built to expressly meet the needs of the knowledge worker. The BACHIMAN/DBA<sup>4</sup> assists in the process of designing databases for DB2. Data from an existing DB2 database can be enhanced using the BACHIMAN/Catalog Extract<sup>4</sup> product in conjunction with the BACHIMAN/DBA and BACHIMAN/Analyst products.

Data can be read, enhanced, and then loaded to the SAA relational database using one or more of the following tools or functions:

- The Data Extract Utility (DXT<sup>1</sup>)
- EDA/Dynamic Extender for DB2 or SQL/DS in conjunction with DXT
- The Data Propagator<sup>1</sup> (DProp<sup>1</sup>)
- Database utilities
- DSS tools or SQL commands, where appropriate
- User-written programs

---

<sup>4</sup> Trademark, Bachman Information Systems, Inc.

<sup>5</sup> Trademark, Computer Associates International, Inc.

## ***Data Extract Utility (DXT)***

DXT provides facilities for data extraction from a variety of sources, including relational and non-relational systems and files on IBM and non-IBM systems. The target database for the extracted data may be:

- A relational database (DB2 or SQL/DS)
- A physical sequential data set or CMS file (which can be routed to other systems)
- Integration Exchange Format (IXF) data set or file (IXF is a data format that can be used for transferring data between programs.)

## ***Data Propagator (DProp)***

Data propagation is the maintenance of two physical copies (hierarchical and relational) as a single logical copy of data and reflecting changes from one to the other. Data Propagator takes changes made by an application program to an IMS/ESA<sup>1</sup> database and applies them to a DB2 table based upon user-specified mapping information. The mapping and application of changes are performed transparently to the application, immediately, within the same unit of work (synchronously). In addition, an interface for user-written programs is provided for invoking the Data Propagator outside of that unit of work (asynchronously).

DProp facilitates the movement of IMS/ESA DB data to DB2 when the periodic movement of data to the relational environment is not sufficient to meet the needs of the user. Creating a separate copy of the IMS/ESA DB data in the form of DB2 tables for use by the knowledge worker can help protect the critical on-line transaction processing application from potential performance degradation resulting from ad-hoc queries contending for the same database resources.

Additional information about DXT and DProp is available in the documentation for those products.

## Positioning

Within the Information Warehouse framework, the SAA SQL database interface is used to access enterprise data. It provides full read/write access to local data managed by any of the four SAA database managers and, via DRDA, to remote data managed by DB2, SQL/DS, and OS/400. It is used for direct access whenever the data is resident on one of the SAA database management systems. It is also used for direct access to non-IBM relational data that is available via DRDA.

If the data is not managed by an SAA database manager or available through DRDA, the data is accessed through the SAA database interface in conjunction with an EDA/Dynamic Extender, if one is available. As of this writing extenders are available for DB2 and SQL/DS. Use of an EDA/Dynamic Extender is intended for existing or new applications coded with SQL statements for DB2 or SQL/DS but wishing to access data not normally available through the SAA SQL database interface. The SAA SQL database interface, in conjunction with the EDA/Dynamic Extender, covers a very significant portion of the customer requirement to access heterogeneous data via a single programming interface.

There may be cases where an application developer may choose to use the EDA API/SQL callable interface. Specifically, the EDA API/SQL might be used when the application executes in a system where an EDA/Dynamic Extender is not yet available, or if the application requires some specialized function which has no equivalent in the SAA SQL database interface (such as converting the returned data to a specific format intended for a specific class of tools).

IBM relational database management systems provide support for optimized data access, transaction integrity, restart and recovery, set processing, and the high performance and availability needed in both local and distributed production environments. Since relational technology is best for addressing the needs of the knowledge worker, the preferred environment and access language for Decision Support System applications is always a relational database via the SAA SQL database interface.

Customers need to give serious consideration to the impact of having Application and Decision Support System users access the operational data directly. As mentioned in the introduction section of this document, the usability of the data in its "native" form, the impact on the transaction systems, and the subset of function supported by EDA/SQL may make data movement to a relational platform more desirable. The movement of data from relational and non-relational sources can be performed using tools that are part of the Information Warehouse framework.

## Notes on the Diagram

The diagram on page 8 illustrates the elements and positioning of Enterprise Data, Data Delivery and Decision Support System applications within the Information Warehouse framework. The path selected for a given application depends upon the specific need being addressed.

### Access to Relational Data (Paths 1 & 2)

Relational data in the framework is accessed directly via the SAA SQL database interface (Path #1 on the diagram). The SQL statements are sent to the relational database management system where they are executed. Requests for local or remote data within the DRDA network are sent to the appropriate system, and the data is returned to the requestor.

When an application or Decision Support System uses an EDA/Dynamic Extender, all of the application's SQL statements are intercepted by the EDA/Dynamic Extender. If the statements are not intended for an EDA/SQL data source, the statements are passed to the local relational database manager (Path #2 on the diagram) without change and with minimal performance impact.

### Access to Non-Relational Data (Path 3)

Read-only access to non-relational data (IMS/DB, VSAM, DEC<sup>6</sup> systems, etc.) is done via the SAA SQL database interface using the subset of embedded dynamic syntax SQL statements which are handled by the EDA/Dynamic Extender for DB2 or SQL/DS. The Dynamic Extender adds the statements needed to input the SQL statement to the EDA/SQL server for processing (Path #3 on the diagram). EDA/SQL translates the SQL statement into the database-specific language needed to access the data, and processes it through one of the available data drivers. The data is returned to the requestor in tabular form.

Relational databases that do not provide distributed database function based on DRDA may also be accessed by using the EDA/Dynamic Extender for DB2 or SQL/DS and the ANSI 89 Level 1 dynamic SQL subset for read-only operations via the appropriate data driver.

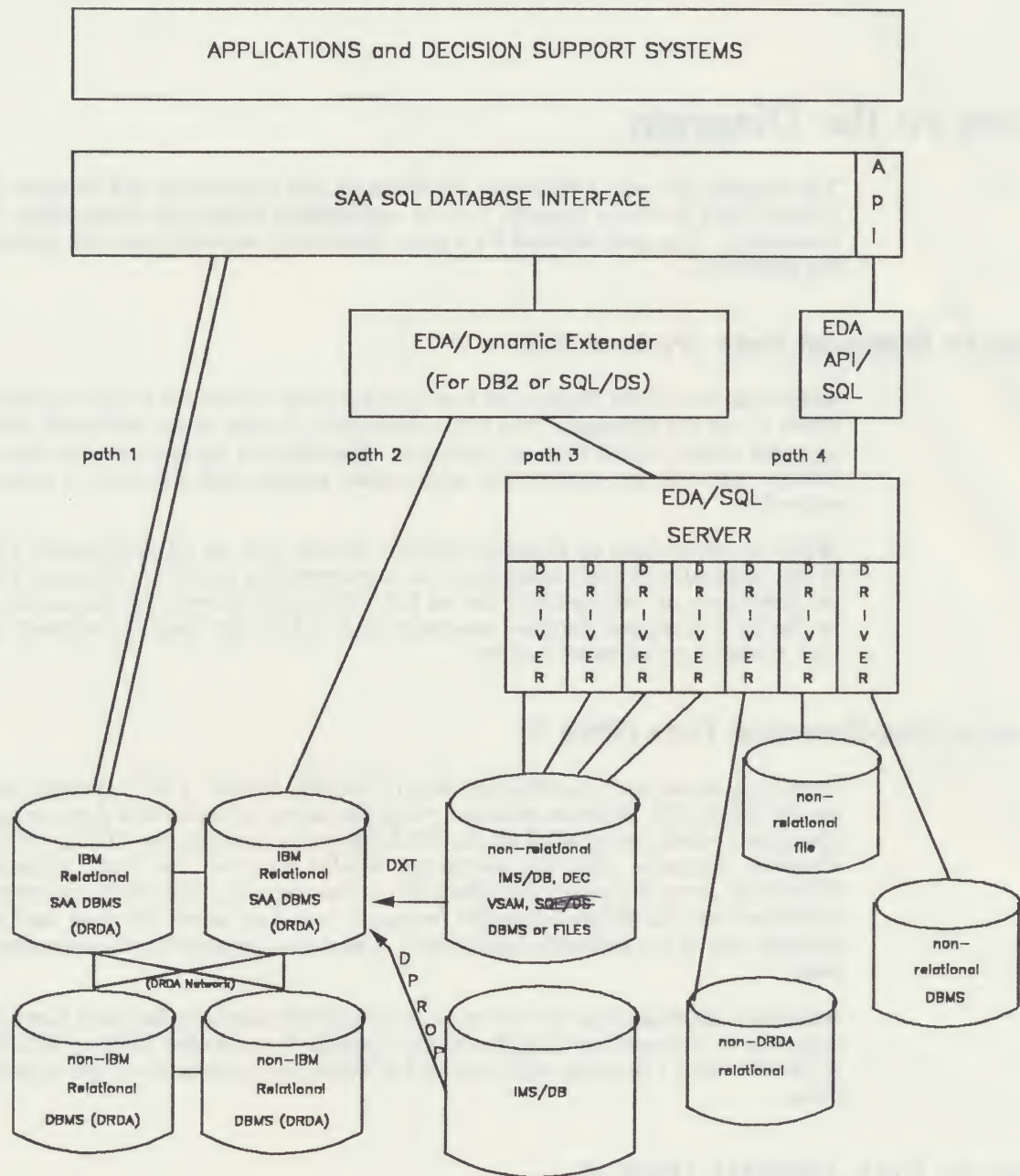
### Access via EDA API/SQL (Path 4)

A customer might consider coding an application with EDA API/SQL if the application or decision support tool used by the knowledge worker needs functions that are not supported by the EDA/Dynamic Extender.

---

<sup>6</sup> Trademark, Digital Equipment Corporation

# INFORMATION WAREHOUSE FRAMEWORK DATA DELIVERY



## Summary

Within the Information Warehouse framework, access to both relational and non-relational data is provided via the SAA SQL database interface. The preferred data store for informational data is a relational database that supports the SAA database interface and DRDA since the relational data structure and the power of the SQL language yield the greatest benefits to the knowledge worker.

The SAA SQL database interface, in conjunction with the EDA/Dynamic Extender (for DB2 or SQL/DS) can provide the users with read-only access to non-relational data via the Decision Support System tools they may already use (such as QMF and AS).

The callable API available for EDA/SQL (EDA API/SQL) might be used when an application executes in a system where an EDA/Dynamic Extender is not yet available, or when the application requires a specialized function which has no equivalent in the SAA SQL database interface (like converting data to a specific format).

Data enhancement and movement to the relational database environment provides the customer the needed data, in a usable form, and a wide variety of Decision Support System product choices.